Lansuite

Contents

Package lansuite_core Procedural Elements

class_debug.php
Containerfile for Debug-Class

- **Package** lansuite_core
- **Author** bytekilla@synergy-lan.de
- **Version** 0.02
- **Filesource** [Source Code for this file](#)

void function d() *[line [24](#)]*

Magic Debugfunction

Calls addvar from Class debug. You can use it 2 Ways. Just pass a Variable or description+Variable. See Example. Caution : for set a Variablecaption like $variable you must User single quote. ('$variable') Example :

```
1    // Just an Variable. Debugoutput generate an automatic Name lika Debugvar_x
2    d($config);
3    // Or Name plus Variabe. Use single quote
4    d('$config', $config);
```

class_framework.php

- **Package** lansuite_core
- **Filesource** [Source Code for this file](#)

Package lansuite_core Classes

Class debug
[line 62]

Debugclass to provide functions like timer, debugvar output, servervar output

Mode : 0 = off, 1 = just generate HTML, 2 = Generate HTML & Write to File By use the Filemode you have do protect the outputdirectory example :

```
1    $debug = new debug(1);
2    $debug->   tracker("BEFOREINCLUDE"          );                // Set Timerpoint
3    $debug->   addvar('$cfg Serverconfig',$cfg);    // Add an Debugvar (Arrays posible)
4    $debug->   timer_start('function sortarray');
5    $array = sortarray($array)
6    $debug->   timer_stop('function sortarray');
7    echo $sys_debug->   show();                      // Show() generates simple HTML-Output
```

- **Package** lansuite_core
- **Author** bytekilla@synergy-lan.de
- **Version** 0.02
- **TODO** Solve Different Configoptions ($config/$cfg) (index.php problem)
- **TODO** Add Procentual Display for Tracker/Timer
- **TODO** Improve HTML-Output (Use just 1 Funktion for better overview)
- **TODO** Improve File-Output (scheiben zur Laufzeit fuer besseres Debuging)
- **TODO** Add Start/Stopp-Timer

Constructor *void* function debug::debug([$mode = "0"], [$debug_path = ""]) *[line   85]*
   Function Parameters:

- *string* **$mode** Debugmode (0=off, 1=normal, 2=file)
- *string* **$debug_path** Path for Filedebug

Constructor Debug-Class Sets Debugmode and Path for Filedebug

void function debug::addvar($key, $value) *[line 160]*
  Function Parameters:

- *string* **$key** Name of the Variable
- *mixed* **$value** Variable

Add Userdefined Debugvariable e.g. addvar('$anz',$anz)
Caution : Use single quote vor Varcaption like $var

void function debug::query_start($query) *[line 202]*
  Function Parameters:

- *string* **$query** Executet Querystring

Start Timer for Querys. Always use with query_stop()

void function debug::query_stop([$error = null]) *[line 214]*
  Function Parameters:

- *string* **$error** Executet Querystring

Stop Timer for Querys. Always use with query_start()

string function debug::show() *[line 329]*
Generating Debug-Table (Simple HTML)

- **TODO** Make Funktions for automatic add/generate Sections ala Timer, Vars, etc.
- **TODO** Add "Jump Top" Links

string function debug::timer_all() *[line 147]*
Print overall time for Timer

- **TODO** Not exact, generate on Call

string function debug::timer_show() *[line 136]*
Print Timer (HTML)

- **Deprecated**

void function debug::timer_start($caption) *[line 180]*
Function Parameters:

- *string* **$caption** String to Identify Timer

Start Timer for testing. Always use with timer_stop()
Shows measured Time in Debugoutput. You can alwas only start and stop just 1 time
Example :
```
1    $debug->   timer_start('function sortarray');
2    $array = sortarray($array)
3    $debug->   timer_stop('function sortarray');
```

- **TODO** Write it

void function debug::timer_stop($caption) *[line 192]*
Function Parameters:

- *string* **$caption** String to Identify Timer

Stop Timer for testing. Always use with timer_start()
For Instructions see timer_start())

- **TODO** Write it

void function debug::tracker($event) *[line 99]*
Function Parameters:

- *string* **$event** Name of Event

Set Timerpoint for Debugoutput. Shows Memoryusage also, if available

Class framework
[line 11]
Manage the Framework/HTML output for Lansuite

- **Package** lansuite_core
- **Author** bytekilla,knox,...
- **Version** Id
- **Access** public

Constructor *void* function framework::framework() *[line 41]*
CONSTRUCTOR : Initialize basic Variables

void function framework::AddToPageTitle($add) *[line 191]*
Function Parameters:

- **$add**

void function framework::add_content($content) *[line [89](#)]*
Function Parameters:

- *string* **$content** Contentstring

Add String/Html to MainContent

void function framework::add_css_code($csscode) *[line [117](#)]*
Function Parameters:

- *string* **$csscode** CSS-Codestring

Add CSS-Code for implementing in Header

void function framework::add_css_path($csspath) *[line [127](#)]*
Function Parameters:

- *string* **$csspath** JS-Codestring

Add Path vor CSS-File for implementing in Header

void function framework::add_js_code($jscode) *[line [98](#)]*
Function Parameters:

- *string* **$jscode** JS-Codestring

Add JS-Code for implementing in Header

void function framework::add_js_path($jspath) *[line [108](#)]*
Function Parameters:

- *string* **$jspath** Path to JS-File

Add JS-File for implementing in Header (as Sourcefile)

string function framework::check_optimizer() *[line 150]*

Check for errors in content and returns Zip-Mode

void function framework::fullscreen($fullscreen) *[line 173]*
Function Parameters:

- *string* **$fullscreen** Fullscreenparameter (yes, no)

Switch Fullscreen-Setting

string function framework::get_clean_url_query($mode) *[line 187]*
Function Parameters:

- *string* **$mode** Needed part of URL (keys : query, base)

Show clean URL-Query for build internal Links

int function framework::get_send_size() *[line 164]*
FÃ¼r Statistik

string function framework::html_out() *[line 205]*
Display/output all HTML new Version

mixed function framework::out_work() *[line 136]*
Calculate the scripts runtime

void function framework::set_design($design) *[line 71]*
Function Parameters:

- *string* **$design** Chosen Design

Set Design

void function framework::set_modus($modus) *[line 80]*
Function Parameters:

- *string* **$modus** Displaymodus (popup,base,print)

Set Displaymodus

Appendices

Appendix A - Class Trees

Package lansuite_core

debug

- debug

framework

- framework

Appendix C - Source Code

Package lansuite_core

File Source for class_debug.php

Documentation for this file is available at class_debug.php

```php
1     <?php
2     /**
3      * Containerfile for Debug-Class
4      *
5      * @author   bytekilla@synergy-lan.de
6      * @version 0.02
7      * @package lansuite_core
8      */
9
10    /**
11     * Magic Debugfunction
12     * Calls addvar from Class debug. You can use it 2 Ways. Just pass a Variable
13     * or description+Variable. See Example. Caution : for set a Variablecaption
14     * like $variable you must User single quote. ('$variable')
15     * Example :
16     * <code>
17     * // Just an Variable. Debugoutput generate an automatic Name lika Debugvar_x
18     * d($config);
19     * // Or Name plus Variabe. Use single quote
20     * d('$config', $config);
21     * </code>
22     *
23     */
24    function d() {
25        global $debug;
26        $arg_count = func_num_args();
27        $arg_vars = func_get_args();
28        if ($debug->  mode >   0) {
29            if ($arg_count==2) {
30                $debug->  addvar($arg_vars[0], $arg_vars[1]);
31            } elseif ($arg_count==1) {
32                $debug->  addvar(NULL, $arg_vars[0]);
33            }
34        }
35    }
36
37    /**
38     * Debugclass to provide functions like timer, debugvar output, servervar output
39     *
40     * Mode : 0 = off, 1 = just generate HTML, 2 = Generate HTML & Write to File
41     * By use the Filemode you have do protect the outputdirectory
42     * example :
43     * <code>
44     * $debug = new debug(1);
45     * $debug->tracker("BEFOREINCLUDE");          // Set Timerpoint
46     * $debug->addvar('$cfg Serverconfig',$cfg);   // Add an Debugvar (Arrays posible)
47     * $debug->timer_start('function sortarray');
48     * $array = sortarray($array)
49     * $debug->timer_stop('function sortarray');
50     * echo $sys_debug->show();                    // Show() generates simple HTML-Output
51     * </code>
52     *
53     * @author   bytekilla@synergy-lan.de
54     * @version 0.02
55     * @package lansuite_core
56     * @todo Add Procentual Display for Tracker/Timer
57     * @todo Solve Different Configoptions ($config/$cfg) (index.php problem)
58     * @todo Add Start/Stopp-Timer
59     * @todo Improve HTML-Output (Use just 1 Funktion for better overview)
60     * @todo Improve File-Output (scheiben zur Laufzeit fuer besseres Debuging)
61     */
62    class debug {
63
64        /**#@+
65         * Intern Variables
66         * @access private
67         * @var mixed
```

```
68             */
69             var $timer_first = ""           ;     // Helpvar Timer
70             var $timer_last = ""            ;      // Helpvar Timer
71             var $timer_out = ""             ;        // Helpvar Timer (Outputstring)
72             var $debugvars = array(); // Uservars to show
73             var $mode = ""          ;              // Debugmode
74             var $debug_path = ""            ;       // Debugpath for Filedebug
75             /**#@-*/
76
77        /**
78         * Constructor Debug-Class
79         * Sets Debugmode and Path for Filedebug
80         *
81         * @param string Debugmode (0=off, 1=normal, 2=file)
82         * @param string Path for Filedebug
83         * @return void
84         */
85         function debug($mode = "0"            , $debug_path = ""             ) {
86             // TODO : argvars verwenden um einfache Variablenübergabe zu ermöglischen.
87             // TODO : Debugbacktrace
88             $this->   mode = $mode;
89             $this->   debug_path = $debug_path;
90             $this->   tracker("INIT DEBUG-CLASS"            );  // Sets first Timerpoint
91         }
92
93
94        /**
95         * Set Timerpoint for Debugoutput. Shows Memoryusage also, if available
96         *
97         * @param string Name of Event
98         */
99         function tracker($event) {
100            if ($this->   mode >    0) {
101                $time = array_sum(explode(" "            , microtime()));
102                // Prepare for Memory, just if functions avail
103                if(function_exists('memory_get_usage')) {
104                    $mem = sprintf("MemAct: %05d Kb   "                    ,
memory_get_usage()/1024);
105                } else {
106                    $mem = ""            ;
107                }
108                if(function_exists('memory_get_peak_usage')) {
109                    $memmax =sprintf("MemPeak: %05d Kb   "                   ,
memory_get_peak_usage()/1024);
110                } else {
111                    $memmax = ""            ;
112                }
113                if(!$this->   timer_first OR $event === false) {
114                    $this->   timer_first = $time;
115                }
116                if(!$this->   timer_last OR $event === false) {
117                    $this->   timer_last = $time;
118                }
119
120                $tmp_out = sprintf("Step:%07.1f ms   Total:%07.1f ms
  "            .$mem.$memmax." => [%s]<br />\n"                           ,
121                        ($time - $this->   timer_last)*1000,
122                        ($time - $this->   timer_first)*1000,
123                        $event);
124
125            $this->   timer_out .= $tmp_out;
126            $this->   timer_all = $time - $this->   timer_first;
127            $this->   timer_last = $time;
128            }
129        }
130
131        /**
132         * Print Timer (HTML)
133         * @return string HTML-Output Timer
134         * @deprecated
135         */
136        function timer_show() {
137            if ($this->   mode >    0) {
138                return $this->   timer_out;
139            }
140        }
141
142        /**
143         * Print overall time for Timer
144         * @return string HTML-Output overall time
```

```php
145          * @todo Not exact, generate on Call
146          */
147         function timer_all() {
148             if ($this->  mode >    0) {
149                 return sprintf("%8.4f"             , $this->   timer_all);
150             }
151         }
152
153         /**
154          * Add Userdefined Debugvariable e.g. addvar('$anz',$anz)
155          * Caution : Use single quote vor Varcaption like $var
156          * @param string  Name of the Variable
157          * @param mixed   Variable
158          */
159         //
160         function addvar($key, $value){
161             if ($this->  mode >    0) {
162                 if (is_string($key)) $this->   debugvars[$key] = $value;
163                     else $this->   debugvars["debugvar_"          .count($this->   debugvars)] = $value;
164             }
165         }
166
167         /**
168          * Start Timer for testing. Always use with timer_stop()
169          * Shows measured Time in Debugoutput. You can alwas only start and stop just 1 time
170          * Example :
171          * <code>
172          * $debug->timer_start('function sortarray');
173          * $array = sortarray($array)
174          * $debug->timer_stop('function sortarray');
175          * </code>
176          * @param string  String to Identify Timer
177          * @todo Write it
178          */
179         //
180         function timer_start($caption){
181             if ($this->  mode >    0) {
182                 // Diverse Ueberpruefungen, evtl mehrfach start/stop erlauben
183             }
184         }
185
186         /**
187          * Stop Timer for testing. Always use with timer_start()
188          * For Instructions see timer_start())
189          * @param string  String to Identify Timer
190          * @todo Write it
191          */
192         function timer_stop($caption){
193             if ($this->   mode >    0) {
194             }
195         }
196
197         /**
198          * Start Timer for Querys. Always use with query_stop()
199          * @param string  Executet Querystring
200          */
201         //
202         function query_start($query){
203             if (($this->   mode >    0) AND (!$this->   sql_query_running)) {
204                 $this->   sql_query_running = true;
205                 $this->   sql_query_start = microtime(true);
206                 $this->   sql_query_string = $query;
207             }
208         }
209
210         /**
211          * Stop Timer for Querys. Always use with query_start()
212          * @param string  Executet Querystring
213          */
214         function query_stop($error = null){
215             if (($this->   mode >    0) AND ($this->   sql_query_running)) {
216                 $this->   sql_query_running = false;
217                 $sql_query_end = microtime(true);
218                 $this->   sql_query_list[] = array(round(($sql_query_end - $this->   sql_query_start)
*1000, 4),$this->   sql_query_string, $error);
219             }
220         }
221
222         /**
223          * Generate and Sort Querylist
```

```php
        * @return array Sortet HTML-Querylist
        * @access private
        */
        function query_fetchlist(){
            if (($this->  mode >    0) AND is_array($this->   sql_query_list)) {
                $this->  sql_query_list = $this->  sort_array_by_col($this->  sql_query_list);
                foreach($this->  sql_query_list as $debug_query) {
                    $sql_query_debug .= debug::row_double(sprintf("<b>%8.4f
ms</b>"      , $debug_query[0]), $debug_query[1]);
                    if (!($debug_query[2]==""          )) $sql_query_debug .=
debug::row_double(""          , "<span style=\"color:red\"><b>Error :
"     .$debug_query[2]."</b></span>"               );
                }
                return $sql_query_debug;
            }
        }

        /**
        * Sort Array by first Column
        * @param  array Array with Columns
        * @return array Sortet Array
        * @access private
        */
        function sort_array_by_col($array){
            function compare($wert_a, $wert_b) {
                // Sortierung nach dem zweiten Wert des Array (Index: 1)
                $a = $wert_a[0];
                $b = $wert_b[0];
                if ($a == $b) {
                    return 0;
                }
                return ($a >    $b) ? -1 : +1;
            }
            usort($array, 'compare');
            return $array;
        }

        /**
        * Generate table heading (HTML)
        * @param  string table heading
        * @return string HTML-Row for table (<tr><td>...</td></tr>)
        * @access private
        */
        function row_top($name){
            $out = "<tr><td width=\"100%\" colspan=\"2\"
bgcolor=\"#C0C0C0\">"               .$name."</td></tr>\n"                    ;
            return $out;
        }

        /**
        * Generate single table row (HTML)
        * @param  string text
        * @return string HTML-Row for table (<tr><td>...</td></tr>)
        * @access private
        */
        function row_single($name){
            $out = "<tr><td width=\"100%\" colspan=\"2\"
align=\"left\">"              .$name."</td></tr>"               ;
            $out .= "<tr><td width=\"100%\" height=\"1\"
bgcolor=\"#C0C0C0\" colspan=\"2\"></td></tr>\n"                        ;
            return $out;
        }

        /**
        * Generate doublerow (HTML)
        * @param  string Keydescription
        * @param  string Variable
        * @return string HTML-Row for Table (<tr><td>...</td></tr>)
        * @access private
        */
        function row_double($key, $value){
            $out = "<tr><td width=\"20%\"
align=\"left\">"              .$key."</td><td width=\"80%\"
align=\"left\">"              .wordwrap( $value, 65, "<br />\n"               , true
)." </td></tr>"              ;
            $out .= "<tr><td width=\"100%\" height=\"1\"
bgcolor=\"#C0C0C0\" colspan=\"2\"></td></tr>\n"                        ;
            return $out;
        }
```

```php
294         /**
295          * Print Array as Tablerows (HTML) Recursive calls posible
296          * @param  string Vararray
297          * @param  string Arraynode (for recursive Calls)
298          * @param  string Levelcounter (for recursive Calls)
299          * @return string HTML-Rows for table (<tr><td>...</td></tr>) multiline
300          * @access private
301          */
302         function row_array($array, $array_node = NULL, $array_level = 0 ){
303             if ($array_level == 0) $out .= debug::row_double("<b>Key</b>"                          ,
"<b>Value</b>"                     );
304             foreach ($array as $key =>    $value) {
305                 $shift = str_repeat("  "                  , $array_level);
306                 if ($array_level==0) {$caption = $key;} else {$caption =
"["          .$key."]"              ;};
307                 // walk types
308                 if (is_array($value)){
309                     $out .=
debug::row_double($shift.$array_node.$caption,"("           .gettype($value).")"          );
310                     $out .= $this->   row_array($value,$array_node.$caption,$array_level+1);
311                 } elseif (is_object($value)) {
312                     $out .= debug::row_double($shift.$array_node.$caption,
"("          .gettype($value).") "             );
313                     $out .= $this-
>  row_array(get_object_vars($value),$array_node.$caption,$array_level+1);
314                 } elseif (is_scalar($value)) {
315                     $out .= debug::row_double($shift.$array_node.$caption,
"("          .gettype($value).") "             .htmlentities($value));
316                 } else {
317                     $out .= debug::row_double($shift.$array_node.$caption,
"("          .gettype($value).") Error: Can not display Debug- Value!!!"          );
318                 }
319             }
320             return $out;
321         }
322
323         /**
324          * Generating Debug-Table (Simple HTML)
325          * @todo Make Funktions for automatic add/generate Sections ala Timer, Vars, etc.
326          * @todo Add "Jump Top" Links
327          * @return string HTML-Table
328          */
329         function show() {
330             if ($this->   mode >    0) {
331                 $this->   tracker("END DEBUG-CLASS"          );
332                 $out .= "<div align=\"left\"><a
name=\"debug_top\"><table width=\"100%\" border=\"0\"
cols=\"0\" cellpadding=\"0\" cellspacing=\"0\">"                          ;
333                 $out .= debug::row_top("<b>Quicknavi</b>"                      );
334                 $out .= debug::row_single("<a
href=\"#debugtracker\">Debugtracker</a> | "
335                                         ."<a href=\"#debugvars\">User-
Debugvars</a> | "
336                                         ."<a
href=\"#post\">\$_POST</a> | "
337                                         ."<a
href=\"#get\">\$_GET</a> | "
338                                         ."<a
href=\"#cookie\">\$_COOKIE</a> | "
339                                         ."<a
href=\"#session\">\$_SESSION</a> | "
340                                         ."<a
href=\"#server\">\$_SERVER</a> | "
341                                         ."<a
href=\"#files\">\$_FILES</a> | "
342                                         ."<a href=\"#sql_querys\">SQL-
Querys</a>"          );
343                 $out .= debug::row_top("<a
name=\"debugtracker\"><b>Debugtracker</b></a>"                          );
344                 $out .= debug::row_single($this->   timer_show());
345                 $out .= debug::row_top("<a name=\"debugvars\"><b>Userdefined
Debugvars</b></a>"              );
346                 $out .= debug::row_array($this->   debugvars);
347                 $out .= debug::row_top("<a name=\"post\"><b>\$_POST-
Data</b></a>"              );
348                 $out .= debug::row_array($_POST);
349                 $out .= debug::row_top("<a name=\"get\"><b>\$_GET-
Data</b></a>"              );
350                 $out .= debug::row_array($_GET);
351                 $out .= debug::row_top("<a name=\"cookie\"><b>\$_COOKIE-
```

```
Data</b></a>"                    );
352          $out .= debug::row_array($_COOKIE);
353          $out .= debug::row_top("<a name=\"session\"><b>\$_Session-
Data</b></a>"                    );
354          $out .= debug::row_array($_SESSION);
355          $out .= debug::row_top("<a name=\"server\"><b>\$_SERVER-
Data</b></a>"                    );
356          $out .= debug::row_array($_SERVER);
357          $out .= debug::row_top("<a name=\"files\"><b>\$_FILES-
Data</b></a>"                    );
358          $out .= debug::row_array($_FILES);
359          $out .= debug::row_top("<a name=\"sql_querys\"><b>SQL-Querys
("     .count($this->   sql_query_list).")</b></a>"                    );
360          $out .= $this->   query_fetchlist();
361          $out .= "</table></div>"                    ;
362          // Mode 2 write complete Debugvars to a file. The directory has to be protected.
363          if ($this->   mode == "2"          ) {
364              echo $this->   mode;
365              $file_handle = fopen($this-
>   debug_path."debug_"          .time().".htm"          , "a"          );
366              fputs($file_handle, $out);
367              fclose($file_handle);
368          }
369          return $out;
370      } else {
371          return ""          ;
372      }
373    }
374  }
```

File Source for class_framework.php

Documentation for this file is available at *class_framework.php*

```php
1      <?php
2
3      /**
4       * Manage the Framework/HTML output for Lansuite
5       *
6       * @package lansuite_core
7       * @author bytekilla,knox,...
8       * @version $Id$
9       * @access public
10      */
11     class framework {
12
13       /**#@+
14        * Intern Variables
15        * @access private
16        * @var mixed
17        */
18        var $timer = ""           ;
19        var $timer2 = ""           ;
20        var $send_size = "0"         ;
21        var $content_crc = ""          ;        // Checksum of Content
22        var $content_size = ""           ;        // Size of Content
23
24        var $internal_url_query = array();      // Clean URL-Query (keys : path, query, base)
25        var $design = "simple"          ;        // Design
26        var $modus = ""             ;        // Displaymodus (popup)
27        var $framework_messages = ""           ;        // All Frameworkmessages
28        var $main_content = ""          ;        // Content
29        var $main_header_metatags = ""             ;        // Headercode for Meta Tags
30        var $main_header_jsfiles = ""            ;        // Headercode for JS-Files
31        var $main_header_jscode = ""             ;        // Headercode for JS-Code
32        var $main_header_cssfiles = ""            ;        // Headercode for CSS-Files
33        var $main_header_csscode = ""             ;        // Headercode for CSS-Code
34        var $IsMobileBrowser = false;
35        var $pageTitle = '';
36       /**#@-*/
37
38       /**
39        * CONSTRUCTOR : Initialize basic Variables
40        */
41        function framework() {
42            // Set Script-Start-Time, to calculate the scripts runtime
43            $this->  design = $design;
44            $this->  timer = time();
45            $this->  timer2 = explode(' ', microtime());
46
47            if (isset($_SERVER['REQUEST_URI'])) {
48                $CurentURL = parse_url($_SERVER['REQUEST_URI']);
49
50                // Filter for Query
51                $URLQuery = preg_replace('/[&]?fullscreen=(no|yes)/sUi'    , '', $CurentURL['query']); // Remove Fullscreenvar
52                $this->  internal_url_query['base'] = $CurentURL['path'].'?'.$CurentURL['query']; // Enspricht alter $CurentURLBase;
53                $this->  internal_url_query['query'] = $URLQuery; // Enspricht alter $URLQuery;
54                $this->  internal_url_query['host'] = $CurentURL['host'];
55            }
56            if (!$this->  internal_url_query['host']) $this->  internal_url_query['host'] = $_SERVER['SERVER_NAME'];
57
58            $this->  add_js_path('ext_scripts/jquery-1.3.2.min.js');
59            $this->  add_js_path('ext_scripts/jquery-ui/jquery-ui-1.7.2.custom.min.js');
60            $this->  add_js_path('scripts.js');
61
62            $this->  add_css_path('ext_scripts/jquery-ui/smoothness/jquery-ui-1.7.2.custom.css');
63            $this->  add_css_path('design/style.css');
64        }
```

```php
65
66      /**
67       * Set Design
68       *
69       * @param string Chosen Design
70       */
71       function set_design($design) {
72           $this->   design = $design;
73       }
74
75      /**
76       * Set Displaymodus
77       *
78       * @param string Displaymodus (popup,base,print)
79       */
80       function set_modus($modus) {
81           $this->   modus = $modus;
82       }
83
84      /**
85       * Add String/Html to MainContent
86       *
87       * @param string Contentstring
88       */
89       function add_content($content) {
90           $this->   main_content .= $content;
91       }
92
93      /**
94       * Add JS-Code for implementing in Header
95       *
96       * @param string JS-Codestring
97       */
98       function add_js_code($jscode) {
99           // Wrapt jeden code neu. Evtl. zusammenfassen
100          $this->   main_header_jscode .= "<script
type=\"text/javascript\">\n"                  .$jscode."\n</script>\n"                    ;
101      }
102
103     /**
104      * Add JS-File for implementing in Header (as Sourcefile)
105      *
106      * @param string Path to JS-File
107      */
108      function add_js_path($jspath) {
109          $this->   main_header_jsfiles .= "<script src=\""                  .$jspath."\"
type=\"text/javascript\"></script>\n"                    ;
110      }
111
112     /**
113      * Add CSS-Code for implementing in Header
114      *
115      * @param string CSS-Codestring
116      */
117      function add_css_code($csscode) {
118          // Wrapt jeden code neu. Evtl. zusammenfassen
119          $this->   main_header_csscode .= "<style type=\"text/css\">\n<!--
\n"      .$csscode."\n-->\n</style>\n"                    ;
120      }
121
122     /**
123      * Add Path vor CSS-File for implementing in Header
124      *
125      * @param string JS-Codestring
126      */
127      function add_css_path($csspath) {
128          $this->   main_header_cssfiles .= "<link rel=\"stylesheet\"
type=\"text/css\" href=\""                  .$csspath."\" />\n"                    ;
129      }
130
131     /**
132      * Calculate the scripts runtime
133      *
134      * @return mixed Scriptruntime
135      */
136      function out_work() {
137          /* Aus der Klasse Sitetool. Feine Sache, evtl. aussagekräfigeren Namen
138          vergeben*/
139          $timer = explode(' ', microtime());
140          $worktime = $timer[1] - $this->   timer2[1];
```

```php
141            $worktime += $timer[0] - $this->    timer2[0];
142            return sprintf("%.5f"            , $worktime);
143        }
144
145     /**
146      * Check for errors in content and returns Zip-Mode
147      *
148      * @return string Returns the possible zip-mode
149      */
150        function check_optimizer() {
151            global $PHPErrorsFound, $db;
152
153            if (headers_sent() or connection_aborted() or $PHPErrors or (isset($db) and $db-
>   errorsFound)) return 0;
154            elseif (strpos($_SERVER["HTTP_ACCEPT_ENCODING"            ], 'x-gzip') !== false) return
"x-gzip"        ;
155            elseif (strpos($_SERVER["HTTP_ACCEPT_ENCODING"            ], 'gzip') !== false) return
"gzip"        ;
156            else return 0;
157        }
158
159     /**
160      * Für Statistik
161      *
162      * @return int Returns the size
163      */
164        function get_send_size(){
165            return $this->    send_size;
166        }
167
168     /**
169      * Switch Fullscreen-Setting
170      *
171      * @param string Fullscreenparameter (yes, no)
172      */
173        function fullscreen($fullscreen){
174            if (isset($fullscreen)) {
175                if ($fullscreen == 'yes')   $_SESSION['lansuite']['fullscreen'] = true;
176                elseif ($fullscreen == 'no')    $_SESSION['lansuite']['fullscreen'] = false;
177            }
178        }
179
180
181     /**
182      * Show clean URL-Query for build internal Links
183      *
184      * @param string Needed part of URL (keys : query, base)
185      * @return string Returns the clean URL-Part
186      */
187        function get_clean_url_query($mode){
188            return $this->    internal_url_query[$mode];
189        }
190
191     function AddToPageTitle($add) {
192        global $cfg;
193
194        if ($add) {
195          if ($this->   pageTitle == '') $this->    pageTitle = $add;
196          else $this->   pageTitle .= ' - '. $add;
197        }
198     }
199
200     /**
201      * Display/output all HTML new Version
202      *
203      * @return string Returns the Complete HTML
204      */
205     function html_out() {
206        global $dsp, $templ, $cfg, $db, $lang, $auth, $smarty, $func, $debug;
207        $compression_mode = $this->    check_optimizer();
208
209        ### Prepare Header
210        if ($_GET['sitereload'])
211        $smarty->   assign('main_header_sitereload', '<meta http-equiv="refresh"
content="'     .$_GET['sitereload'].';
URL='.$_SERVER["PHP_SELF"          ].'?'.$_SERVER['QUERY_STRING'].'">'           );
212        // Add special CSS and JS
213        $smarty->   assign('main_header_metatags', $this->   main_header_metatags);
214        $smarty->   assign('main_header_jsfiles', $this->   main_header_jsfiles);
215        $smarty->   assign('main_header_jscode', $this->   main_header_jscode);
```

```
216         $smarty->    assign('main_header_cssfiles', $this->   main_header_cssfiles);
217         $smarty->    assign('main_header_csscode', $this->   main_header_csscode);
218
219         $smarty->    assign('IsMobileBrowser', $this->   IsMobileBrowser);
220         $smarty->    assign('DisplayMode', $this->   modus);
221
222         $smarty->    assign('MainTitle', $this->   pageTitle);
223         $smarty->    assign('MainLogout', '');
224         $smarty->    assign('MainLogo', '');
225         $smarty->    assign('MainBodyJS', $templ['index']['body']['js']);
226         $smarty->    assign('MainJS', $templ['index']['control']['js']);
227         $smarty->    assign('MainContent', $this->   main_content);
228
229         $EndJS = '';
230         if ($cfg['google_analytics_id']) $EndJS = '<script type="text/javascript">
231    var gaJsHost = (("https:" == document.location.protocol) ? "https://ssl." :
"http://www.");
232    document.write(unescape("%3Cscript src=\'" + gaJsHost + "google-
analytics.com/ga.js\' type=\'text/javascript\'%3E%3C/script%3E"));
233    </script>
234    <script type="text/javascript">
235    var pageTracker = _gat._getTracker("'      . $cfg['google_analytics_id'] .'");
236    pageTracker._trackPageview();
237    </script>'        ;
238         $smarty->   assign('EndJS', $EndJS);
239
240         ### Switch Displaymodus (popup, base, print, normal)
241         switch ($this->   modus) {
242           case 'print':
243             // Make a Printpopup (without Boxes and Special CSS for printing)
244             $smarty->   assign('MainContentStyleID', 'ContentFullscreen');
245             $smarty->   display("design/simple/templates/main.htm"          );
246           break;
247
248           case 'popup':
249             // Make HTML for Popup
250             $smarty->    assign('MainContentStyleID', 'ContentFullscreen');
251
252             // TODO : Rendundant... zusammenfassen
253             if ($compression_mode and $cfg['sys_compress_level']) {
254                 header("     Content-Encoding: $compression_mode"     );
255                 echo "\x1f\x8b\x08\x00\x00\x00\x00\x00"         ;
256                 $index = $smarty->   fetch("   design/{$this->   design}/templates/main.htm"). "\n<!-
- SiteTool - Compressed by $compression_mode -->   ";
257                 $this->   content_size = strlen($index);
258                 $this->   content_crc = crc32($index);
259                 $index = gzcompress($index, $cfg['sys_compress_level']);
260                 $index = substr($index, 0, strlen($index) - 4); // Letzte 4 Zeichen werden abgeschnitten.
Aber Warum?
261                 echo $index;
262                 echo pack('V', $this->   content_crc) . pack('V', $this->   content_size);
263             } else $smarty->   display("design/{$this->   design}/templates/main.htm");
264           break;
265
266           case 'base':
267             // Make HTML for Sites Without HTML (e.g. for generation Pictures etc)
268             echo $this->   main_content;
269           break;
270
271             case 'ajax':
272             // Make HTML for Sites Without HTML (e.g. for generation Pictures etc)
273             echo $this->   main_content;
274           break;
275
276           default :
277             // Footer
278             $smarty->   assign('main_footer_version', $templ['index']['info']['version']);
279             $smarty->   assign('main_footer_date', date('y'));
280             $smarty->   assign('main_footer_countquery',$db->   count_query);
281             $smarty->   assign('main_footer_timer', round($this->   out_work(), 2));
282             $smarty->   assign('main_footer_cleanquery', $this->   get_clean_url_query('query'));
283
284             $main_footer_mem_usage = '';
285             if (function_exists('memory_get_peak_usage')) $main_footer_mem_usage = 'Memory-Usage: '.
$func->   FormatFileSize(memory_get_peak_usage()) .' |';
286             $smarty->   assign('main_footer_mem_usage', $main_footer_mem_usage);
287
288             $footer = $smarty->   fetch('design/templates/footer.htm');
289
290             if ($cfg["sys_optional_footer"          ]) $footer .=
```

```
       HTML_NEWLINE.$cfg["sys_optional_footer"              ];
291              $smarty->  assign('Footer', $footer);
292
293              // Normal HTML-Output with Boxes
294              $smarty->  assign('Design', $this->  design);
295
296              // Unterscheidung fullscreen / Normal
297              if ($_SESSION['lansuite']['fullscreen']) $smarty->  assign('MainContentStyleID',
       'ContentFullscreen');
298              else $smarty->  assign('MainContentStyleID', 'Content');
299
300              if ($auth['login']) $smarty->  assign('MainLogout', '<a
       href="index.php?mod=auth&action=logout" class="menu">Logout</a>'                            );
301
302              // Ausgabe Hauptseite
303              if (!$_SESSION['lansuite']['fullscreen']) {
304                $smarty->  assign('MainFrameworkmessages', $this->  framework_messages);
305                $smarty->  assign('MainLeftBox', $templ['index']['control']['boxes_letfside']);
306                $smarty->  assign('MainRightBox', $templ['index']['control']['boxes_rightside']);
307                $smarty->  assign('MainLogo', '<img src="design/'        .$this-
       >  design.'/images/lansuite-logo.gif" alt="Lansuite Logo" title="Lansuite Logo"
       border="0" />'            );
308                if ($auth['type'] >=   2 and $cfg['sys_showdebug']) {
309                  $smarty->  assign('MainDebug', $debug->  show());
310                }
311              } else {
312                // Ausgabe Vollbildmodus
313                $smarty->  assign('CloseFullscreen', '<a href="index.php?'       . $this-
       >  get_clean_url_query('query') .'&amp;fullscreen=no" class="menu"><img
       src="design/'     . $this->  design .'/images/arrows_delete.gif" border="0"
       alt="" /><span class="infobox">'                              . t('Vollbildmodus schlieÃŸen)
       .'</span> Lansuite - Vollbildmodus</a>'            );
314              }
315
316              // Start Javascript-Code for MainContent with JQuery-Tabs
317              /*$this->main_header_jscode .= "
318                  $(document).ready(function(){
319                    $('#MainContentTabs').tabs({
320                      click: function(tab) {
321                        location.href = $.data(tab, 'href');
322                        return false;
323                      }
324                    });
325                  });
326              ";*/
327
328              // MainContent with JQuery-Tabs for LS-Messenger
329              #$main_content_with_tabs .= "<div class='ui-tabs ui-widget ui-widget-content ui-
       corner-all' id='MainContentTabs'>\n";
330              #$main_content_with_tabs .= "  <ul class='ui-tabs-nav ui-helper-reset ui-helper-
       clearfix ui-widget-header ui-corner-all'>\n";
331              #$main_content_with_tabs .= "    <li class='ui-state-default ui-corner-top ui-
       tabs-selected ui-state-active'><a href='#main_content'
       title='Lansuite'><em>Lansuite</em></a></li>\n";
332              #$main_content_with_tabs .= "  </ul>\n";
333              #$main_content_with_tabs .= "  <div class='ui-content'>\n";
334              #$main_content_with_tabs .= "    <div id='main_content'>\n";
335              #$main_content_with_tabs .= "    <br />\n";
336              #$main_content_with_tabs .= $this->main_content;
337              #$main_content_with_tabs .= "    </div>\n";
338              #$main_content_with_tabs .= "  </div>\n";
339              #$main_content_with_tabs .= "</div>\n";
340
341              #$smarty->assign("MainContent", $main_content_with_tabs);
342
343            // Ausgabe des Hautteils mit oder ohne Kompression
344            if ($compression_mode and $cfg['sys_compress_level']) {
345              header("Content-Encoding: $compression_mode");
346              echo "\x1f\x8b\x08\x00\x00\x00\x00\x00"          ;
347              $index = $smarty->  fetch("design/{$this->  design}/templates/main.htm") ."\n<!--
       SiteTool - Compressed by $compression_mode -->   ";
348              $this->  content_size = strlen($index);
349              $this->  content_crc = crc32($index);
350              $index = gzcompress($index, $cfg['sys_compress_level']);
351              $index = substr($index, 0, strlen($index) - 4); // Letzte 4 Zeichen werden abgeschnitten.
       Aber Warum?
352              echo $index;
353              echo pack('V', $this->  content_crc) . pack('V', $this->  content_size);
354            } else $smarty->  display("design/{$this->  design}/templates/main.htm");
355          break;
```

```
356        }
357      }
358    }
```

Appendix D - Todo List

In Package lansuite_core

In **debug**

- Add Procentual Display for Tracker/Timer

- Add Start/Stopp-Timer

- Improve File-Output (scheiben zur Laufzeit fuer besseres Debuging)

- Improve HTML-Output (Use just 1 Funktion for better overview)

- Solve Different Configoptions ($config/$cfg) (index.php problem)

In **debug::show()**

- Add "Jump Top" Links

- Make Funktions for automatic add/generate Sections ala Timer, Vars, etc.

In **debug::timer_all()**

- Not exact, generate on Call

In **debug::timer_start()**

- Write it

In **debug::timer_stop()**

- Write it

Index